APPROXIMATION ALGORITHMS FOR MIXED FRACTIONAL PACKING AND COVERING PROBLEMS [*]

Klaus Jansen, [1]

[1] *University of Kiel*
Olshausenstr. 40, 24118 Kiel, Germany
kj@informatik.uni-kiel.de

Abstract   We propose an approximation algorithm based on the Lagrangian or price - directive decomposition method to compute an ϵ-approximate solution of the mixed fractional packing and covering problem: find $x \in B$ such that $f(x) \leq (1 + \epsilon)a$, $g(x) \geq (1 - \epsilon)b$ where $f(x), g(x)$ are vectors with M nonnegative convex and concave functions, a and b are M - dimensional nonnegative vectors and B is a convex set that can be queried by an optimization or feasibility oracle. We propose an algorithm that needs only $O(M\epsilon^{-2} \ln(M\epsilon^{-1}))$ iterations or calls to the oracle. The main contribution is that the algorithm solves the general mixed fractional packing and covering problem (in contrast to pure fractional packing and covering problems and to the special mixed packing and covering problem with $B = \mathbb{R}_+^N$) and runs in time independent of the so-called width of the problem.

Keywords: Convex and concave optimization, approximation algorithms.

1   Introduction.

We study mixed fractional packing and covering problems (MPC_ϵ) of the following form: Given a vector $f : B \to \mathbb{R}_+^M$ of M nonnegative continuous convex functions and a vector $g : B \to \mathbb{R}_+^M$ of M nonnegative continuous concave functions, two M - dimensional nonnegative vectors a, b, a nonempty convex compact set B and a relative tolerance $\epsilon \in (0, 1)$, find an approximately feasible vector $x \in B$ such that $f(x) \leq (1 + \epsilon)a$ and $g(x) \geq (1 - \epsilon)b$ or find a

proof that no vector is feasible (that satisfies $x \in B$, $f(x) \leq a$ and $g(x) \geq b$). W.l.o.g. we may assume that a and b are equal to the vector e of all ones.

The fractional packing problem with convex constraints, i.e. to find $x \in B$ such that $f(x) \leq (1 + \epsilon)a$, is solved in [6, 7, 10] by the Lagrangian decomposition method in $O(M(\epsilon^{-2} + \ln M))$ iterations where each iteration requires a call to an approximate block solver $ABS(p, t)$ of the form: find $\hat{x} \in B$ such that $p^T f(\hat{x}) \leq (1 + t)\Lambda(p)$ where $\Lambda(p) = \min_{x \in B} p^T f(x)$. Furthermore, Grigoriadis et al. [8] proposed also an approximation algorithm for the fractional covering problem with concave constraints, i.e. to find $x \in B$ such that $g(x) \geq (1 - \epsilon)b$, within $O(M(\epsilon^{-2} + \ln M))$ iterations where each iteration requires here a call to an approximate block solver $ABS(q, t)$ of the form: find $\hat{x} \in B$ such that $q^T g(\hat{x}) \geq (1 - t)\Lambda(q)$ where $\Lambda(q) = \max_{x \in B} q^T g(x)$. Both algorithms solve also the corresponding min-max and max-min optimization variants within the same number of iterations. Furthermore, the algorithms can be generalized to the case where the block solver has arbitrary approximation ratio [9–11].

Further interesting algorithms for the fractional packing and fractional covering problem with linear constraints were developed by Plotkin et al. [14] and Young [16]. These algorithms have a running time that depends linearly on the width ρ - an unbounded function of the input instance. Several relatively complicated techniques were proposed to reduce this dependence. Garg and Könemann [5] described a nice algorithm for the fractional packing problem with linear constraints that needs only $O(M\epsilon^{-2} \ln M)$ iterations. On the other hand, the algorithm by Grigoriadis et al. [8] is the only known algorithm that solves the fractional covering problem with a number of iterations independently on the width.

For the mixed packing and covering problem (with linear constraints and polytope B), Plotkin et al. [14] proposed also approximation algorithms where the running time depends on the width. They present an algorithm that uses $O(M^2(\ln^2 \rho)\epsilon^{-2} \ln(\epsilon^{-1}M \ln \rho) \ln \rho)$ calls to an oracle of the form: find a vertex $\hat{x} \in B$ with $f(\hat{x}) \leq va$ and

$$p^T f(\hat{x}) - \sum_{m \in I(v, \hat{x})} q_m g_m(\hat{x}) = \min\{p^T f(x) - \sum_{m \in I(v, x)} q_m g_m(x) | x \text{ vertex of } B\}$$

where $I(v, x) = \{m | g_m(x) \leq vb_m\}$, v is a constant and where

$$\rho = \max_{x \in B} \max_{m=1,...,M} (f_m(x)/a_m, g_m(x)/b_m).$$

Young [17] described an approximation algorithm for a special mixed packing and covering problem with linear constraints with non-negative coefficients and special convex set $B = \mathbb{R}_+^N$. The algorithm has a running time of $O(M^2\epsilon^{-2} \ln M)$. Recently, Fleischer [4] gave an approximation scheme

for the optimization variant (minimizing $c^T x$ such that $Cx \geq a$, $x \leq b$ and $x \geq 0$ where a, b, and c are nonnegative integer vectors and C is a nonnegative integer matrix). Applications of the pure and mixed fractional packing and covering problems can be found in [1–5, 9–12, 14, 17]. Young [17] posed the following open problem: find an efficient width-independent Lagrangian-relaxation algorithm for the abstract mixed packing and covering problem: find $x \in B$ such that $Px \leq (1 + \epsilon)a$, $Cx \geq (1 - \epsilon)b$, where P, C are nonnegative matrices, a, b are nonnegative vectors and B is a polytope that can be queried by an optimization oracle (given a vector c, return $x \in B$ minimizing $c^T x$) or some other suitable oracle.

New Result: Our contribution here is an efficient width-independent Lagrangian - relaxation algorithm for the mixed packing and covering problem that uses a suitable optimization oracle of the form (given two vectors c, d, return $x \in B$, $d^T x \geq 1$, minimizing $c^T x$). Interestingly, a feasibility oracle of the form (given two vectors c, d, return $x \in B$ such that $c^T x \leq 1$ and $d^T x \geq 1$) is also sufficient. This solves the open problem by N.E. Young [17]. Interestingly, our algorithm works also for a more general problem with a convex set B and nonnegative convex packing and concave covering constraints.

The algorithm uses a variant of the Lagrangian or price directive decomposition method. This is an iterative strategy that solves (MPC_ϵ) by computing a sequence of triples (p, q, x) as follows. A coordinator uses the current vector $x \in B$ to compute two price vectors $p = p(x) \in \mathbb{R}_+^M$ and $q = q(x) \in \mathbb{R}_+^M$ with $\sum_{m=1}^{M} p_m + q_m = 1$. Then the coordinator calls an optimization oracle to compute a solution $\hat{x} \in B$ of the block problem (BP) of the form $\Lambda(p, q) = \min\{p^T f(y) | y \in B, q^T g(y) \geq \sum_{m=1}^{M} q_m\}$, and makes a move from x to $(1 - \tau)x + \tau\hat{x}$ with an appropriate step length $\tau \in (0, 1)$. Such a iteration is called a coordination step. For our algorithm, we only require an approximate block solver (ABS) that solves the underlying block problem (BP) to a given relative tolerance $t \in (0, 1)$:

$$ABS(p, q, t): \qquad \text{compute} \quad \hat{x} = \hat{x}(p, q) \in B \text{ such that}$$

$$p^T f(\hat{x}) \leq (1 + t)\Lambda(p, q) \text{ and } q^T g(\hat{x}) \geq \tfrac{1}{1+t} \sum_{m=1}^{M} q_m.$$

Our main result is the following:

THEOREM 1 *There is an approximation algorithm that for any given accuracy $\epsilon \in (0, 1)$ solves the mixed fractional packing and covering problem (MPC_ϵ) within*

$$N = O(M\epsilon^{-2} \ln(M\epsilon^{-1}))$$

iterations or coordination steps, where each of which requires a call to $ABS(p, q, \Theta(\epsilon))$ and a coordination overhead of $O(M \ln(M\epsilon^{-1}))$ arithmetic operations.

Alternatively, instead of using the approximate block solver an approximate feasibility oracle of the form (compute $\hat{x} \in B$ such that $p^T f(\hat{x}) \leq (1 + t) \sum_{m=1}^{M} p_m$ and $q^T g(\hat{x}) \geq \frac{1}{1+t} \sum_{m=1}^{M} q_m$) is also sufficient.

Main Ideas: Our algorithm builds on approximation schemes of Grigoriadis et al. [7, 8] and Young [17]. One of the ideas is to combine two different potential functions that were proposed for pure fractional packing and covering problems [7, 8]. We associate here with the packing and covering constraints the following potential function:

$$\Phi'_t(\theta, x) = 2 \ln \theta - \frac{t}{M} \sum_{m=1}^{M} \ln(\theta - f_m(x)) - \frac{t}{M} \sum_{m=1}^{M} \ln(g_m(x) - \frac{1}{\theta})$$

where $\theta \in \mathbb{R}_+$ and $t > 0$ is a tolerance that depends on ϵ and is used in the approximate block solver. The function Φ' can be extremely small, since there is no upper bound on the function values $g_m(x)$. Let A be a nonempty subset of $\mathcal{M} = \{1, \dots, M\}$. To control the values of the covering functions $g_m(x)$ and to have a lower bound for the potential function, we eliminate functions g_m (and the corresponding index in A) when the function value $g_m(x)$ is larger than a prespecified threshold value T and modify the potential function. Let $A(x)$ denote the index set corresponding to a given vector $x \in B$. Then the modified potential function has the form:

$$\Phi_t(\theta, x, A(x)) = 2 \ln \theta - \frac{t}{M} \sum_{m=1}^{M} \ln(\theta - f_m(x))$$

$$-\frac{t}{M} \sum_{m \in A(x)} \ln(g_m(x) - \frac{1}{\theta}) - \frac{t}{M} \sum_{m \notin A(x)} \ln(T).$$

The potential function Φ_t has an unique minimum $\theta_{A(x)}(x)$ that approximates the objective value

$$\lambda_{A(x)}(x) = \max(\max_{m \in \mathcal{M}} f_m(x), \max_{m \in A(x)} 1/g_m(x)).$$

This potential function Φ_t and the minimizer $\theta_{A(x)}(x)$ is used to define the price vectors $p = p(x)$ and $q = q(x)$ for the current vector $x \in B$ and to optimize in the correct direction. Another important parameter for the convergence of the algorithm is the reduced potential value $\phi_t(x, A(x)) = \Phi_t(\theta_{A(x)}(x), x, A(x))$ for $x \in B$ and $A(x) \subset \{1, \dots, M\}$. Since we can not control the values of eliminated functions g_m for $m \notin A(x)$ (after the elimination), at the end of each phase s we take a convex combination over different computed vectors.

The main problem is to choose a good step length τ in order to obtain a fast and width-independent convergence. To achieve this goal we study the following four cases:

$$(1) : p^T f(x) - p^T f(\hat{x}) \geq 0 \text{ and } q^T g(\hat{x}) - q^T g(x) \geq 0,$$

$$(2) : p^T f(x) - p^T f(\hat{x}) < 0 \text{ and } q^T g(\hat{x}) - q^T g(x) \geq 0,$$

$$(3) : p^T f(x) - p^T f(\hat{x}) \geq 0 \text{ and } q^T g(\hat{x}) - q^T g(x) < 0,$$

$$(4) : p^T f(x) - p^T f(\hat{x}) < 0 \text{ and } q^T g(\hat{x}) - q^T g(x) < 0,$$

where x is the current solution and $\hat{x}$ is the block solution corresponding to the price vectors $p = p(x)$ and $q = q(x)$. Case (4) with $p^T f(x) - p^T f(\hat{x}) < 0$ and $q^T g(\hat{x}) - q^T g(x) < 0$ is not possible. In this case one of the stopping rules is satisfied and the algorithm stops with the iterate x. The step length τ is defined carefully in dependence on the cases $(1) - (3)$ and the minimizer $\theta_{A(x)}(x)$ of the potential function. In the general case, the coordinator moves from solution x to $(1 - \tau)x + \tau\hat{x}$ and sets the index set $A(x') = \{m \in A(x) | g_m(x') < T\}$. In the case where $\max_{m \in A} g_m(x)(1 - \tau) + g_m(\hat{x})\tau > T$ we reduce the step length from τ to $\bar{\tau}$ and use as next vector $x' = (1 - \bar{\tau})x + \bar{\tau}\hat{x}$. This is important for the convergence analysis.

2 Potential function and price vectors

Let A be a nonempty subset of $\mathcal{M} = \{1, \ldots, M\}$. During a phase, we eliminate a concave function g_m (and the corresponding index in A) when the function value $g_m(x) \geq T$. Let $A(x)$ denote the index set corresponding to a given vector $x \in B$. For simplicity we use $A = A(x)$ (if the dependence is clear).

2.1 Potential function

The potential function Φ_t (given above) is well defined for $\lambda_A(x) < \theta < \infty$ where

$$\lambda_A(x) = \max(\max_{1 \leq m \leq M} f_m(x), \max_{m \in A} \frac{1}{g_m(x)}).$$

If $g_m(x) = 0$ for at least one index $m \in A$ then we define $\lambda_A(x) = \infty$. Furthermore, Φ_t has the barrier property (i.e. $\Phi_t(\theta, x, A) \to \infty$ for $\theta \to \infty$ and for $\theta \to \lambda_A(x)$). We define the reduced potential function $\phi_t(x, A)$ as the minimum value $\Phi_t(\theta, x, A)$ over $\theta \in (\lambda_A(x), \infty)$ for a given $x \in B$. The unique minimizer $\theta_A(x)$ can be determined from the first-order optimality condition:

$$\frac{t\theta}{M} \sum_{m=1}^{M} \frac{1}{\theta - f_m(x)} + \frac{t}{M\theta} \sum_{m \in A} \frac{1}{g_m(x) - 1/\theta} = 2. \qquad (1)$$

The implicit function $\theta_A(x)$ approximates $\lambda_A(x)$. This is important for the further analysis.

LEMMA 2

$$\theta_A(x)/(1 + t/(2M)) \geq \lambda_A(x) \geq \theta_A(x)(1 - \frac{t}{2} - \frac{t|A|}{2M}) \geq \theta_A(x)(1 - t).$$

Lemma 2 shows that the value $\theta_A(x)$ approximates the objective value $\lambda_A(x)$ for small t. Interestingly, the reduced potential function $\phi_t(x, A)$ can be bounded also in terms of $\theta_A(x)$.

LEMMA 3 *If* $g_m(x) \leq T$ *for each* $m \in A$ *then* $\phi_t(x, A) \geq (2 - t) \ln \theta_A(x) - t \ln T$. *Furthermore, if* $T > 1/\lambda_A(x)$ *then* $\phi_t(A, x) \leq 2 \ln \theta_A(x) + 2t \ln(\frac{2M}{t}) + t \ln(1 + t/(2M))$.

2.2 Price vectors

Given a vector $x \in B$ and a subset $A \subset \{1, \dots, M\}$, the price vector $p(x, A)$ is defined by

$$p_m(x, A) = \frac{t}{2M} \frac{\theta_A(x)}{\theta_A(x) - f_m(x)} \qquad (2)$$

and the price vector $q(x, A)$ is given by

$$q_m(x, A) = \begin{cases} \frac{t}{2M} \frac{1}{g_m(x)\theta_A(x) - 1} & m \in A, \\ \\ 0 & \text{otherwise.} \end{cases} \qquad (3)$$

Using the first-order condition, $\sum_{m=1}^{M} p_m(x, A) + \sum_{m=1}^{M} q_m(x, A) = 1$ and each component $p_m(x, A), q_m(x, A)$ is nonnegative.

LEMMA 4 **(a)** $p(x, A)^T f(x) = \theta_A(x)(\sum_{m=1}^{M} p_m(x, A) - t/2) \leq \theta_A(x)(1 - t/2)$,

(b) $q(x, A)^T g(x) = (\sum_{m \in A} q_m(x, A) + t|A|/(2M))/\theta_A(x) \leq (\sum_{m \in A} q_m(x, A) + t/2)/\theta_A(x) \leq (1 + t/2)/\theta_A(x)$.

Notice that Lemma 4 (a) implies that $\sum_{m=1}^{M} p_m(x, A) \geq t/2$.

3 Our approximation algorithm

In this section we describe the approximation algorithms for the mixed fractional packing and covering problem. First we suppose that there exists a feasible solution $x \in B$ with $f(x) \le e$ and $g(x) \ge e$. Then the approximation algorithm works as follows:

(1) compute initial solution $x^{(0)}$, $s := 0$, $\epsilon_0 := 1/4$;

(2) repeat {scaling phase }

    (2.1) $s := s+1$; $\epsilon_s := \epsilon_{s-1}/2$; $x := x^{(s-1)}$; $T(s) := 528(M^3/\epsilon_s^2)/\lambda_{\mathcal{M}}(x)$; $A := \{m \in \{1, \ldots, M\} | g_m(x) < T(s)\}$; $finished := false$; $k := 0$;

    (2.2) if $A \ne \{1, \ldots, M\}$ **then begin** k:=k+1; $x_k := x$ **end**;

    (2.3) if stopping rule 1 is satisfied for x **then** $finished := true$; $y := x$ **end**;

    (2.4) while $not(finished)$ **do begin**

        (2.4.1) compute $\theta_A(x)$, $p(x, A)$ and $q(x, A)$;

        (2.4.2) $\hat{x} := ABS(p(x, A), q(x, A), \epsilon_s/32)$;

        (2.4.3) if one of the stopping rules is satisfied
            then begin $finished := true$; $y := x$ **end**
            else begin
            (2.4.3.1) compute step length τ and $x' := (1 - \tau)x + \tau\hat{x}$;
            (2.4.3.2) if $\max_{m \in A} g_m(x)(1 - \tau) + g_m(\hat{x})\tau > T(s)$ **then** reduce
                τ to $\bar{\tau}$ and $x' := (1 - \bar{\tau})x + \bar{\tau}\hat{x}$;
            (2.4.3.3) $A' := A \setminus \{m | g_m(x') \ge T(s)\}$; $x := x'$;
            (2.4.3.4) if $A \ne A'$ **then begin** $k := k + 1$; $x_k = x'$;
                $A := A'$ **end**

        end

    end;

    (2.5) compute convex combination of $x_1, \ldots, x_k, y$ to get $x^{(s)}$;

    (2.6) until $\epsilon_s \le \epsilon/2$ or $\lambda(x^{(s)}) \le 1 + \epsilon$;

(3) return($x^{(s)}$).

The details of the algorithm are described later in this section (how to compute an initial solution, the stopping rules, the choice of the step length, and the reduction of the step length). For the case where the set of feasible solutions $\{x \in B | f(x) \le e, g(x) \ge e\}$ is empty, we have to modify the program above. If an inequality $p(x, A)^T f(\hat{x}) > (1+t) \sum p_m(x, A)$ holds for a block solution $\hat{x}$, then we can conclude that there is no feasible solution.

3.1 Initial solution

For each $m \in \{1, \ldots, M\}$, we consider the block problem (B_m) of the form $\Lambda(p, q) = \min\{\frac{1}{M} \sum_{\ell=1}^{M} f_\ell(x) | x \in B, g_m(x) \geq 1\}$ where $p = (1/M, \ldots, 1/M)$ and $q = e_m$ is the unit vector with all zero coordinates except for its m.th component which is 1. If there is a solution $\bar{x} \in B$ with $f(\bar{x}) \leq e$ and $g(\bar{x}) \geq e$, then this solution satisfies $(1/M) \sum_{\ell=1}^{M} f_\ell(\bar{x}) \leq 1$ and $g_m(\bar{x}) \geq 1$. Let $\hat{x}^{[m]} \in B$ be an approximate solution of the block problem (B_m) with tolerance $t = 1/2$, and let $x^{(0)} = (1/M) \sum_{m=1}^{M} \hat{x}^{[m]}$. Using the convexity of B, $x^{(0)} \in B$. If the approximate solution satisfies $(1/M) \sum_{\ell=1}^{M} f_\ell(\hat{x}^{[m]}) > 1 + t$, then we conclude that the solution set of the mixed problem is empty. In the other case we can prove:

LEMMA 5 *If there exists a feasible solution of the mixed packing and covering problem, then $\lambda(x^{(0)}) \leq 3M/2$.*

3.2 Stopping rules

In the algorithm we stepwise decrease in phases the objective value λ from $3M/2$ to $1/(1 - \epsilon/2)$. In the first phase we decrease $3M/2$ to $\epsilon_1 = 1/8$. After that we set $\epsilon_s = \epsilon_{s-1}/2$. The goal in phase s is to obtain a solution $x^{(s)}$ with $\lambda(x^{(s)}) \leq 1/(1 - \epsilon_s)$. In order to get such a solution we need at the end of phase s a solution y with $\lambda_A(y) \leq 1/(1 - \epsilon_s/4)$. This is necessary, since we eliminate covering constraints within the phases. To obtain the solution y and to show the convergence we use three stopping rules. For the first rule we simply test whether

$$\lambda_A(x) \leq 1 + \epsilon_s/4 \tag{4}$$

for the current solution x. For this rule we get immediately

LEMMA 6 *If $\lambda_A(x) \leq 1 + \epsilon_s/4$ then $f_m(x) \leq 1 + \epsilon_s/4 \leq 1/(1 - \epsilon_s/4)$ for each $m \in \{1, \ldots, M\}$ and $g_m(x) \geq 1/(1 + \epsilon_s/4) \geq 1 - \epsilon_s/4$ for each $m \in A$.*

For the second rule we define a parameter ν that depends on the current iterate x and the approximate block solution $\hat{x}$ as follows

$$\nu = \nu(x, \hat{x}) = \frac{p^T f(x) - p^T f(\hat{x}) + \theta(q^T g(\hat{x}) - q^T g(x))}{p^T f(x) + p^T f(\hat{x}) + \theta(q^T g(\hat{x}) + q^T g(x))} \tag{5}$$

where $p = p(x, A)$, $q = q(x, A)$ and $\theta = \theta_A(x)$. Clearly, $\nu(x, \hat{x}) \leq 1$. The Lemma below states that x is an approximate solution of the phase s corresponding to subset A, when ν is bounded by $t_s = \Theta(\epsilon_s)$.

LEMMA 7 *Suppose $\epsilon_s \in (0, 1)$ and $t_s = \epsilon_s/32$. For a given $x \in B$, let p, q be computed by $(2, 3)$ and $\hat{x}$ computed by $ABS(p, q, t_s)$. If $\nu(x, \hat{x}) \leq t_s$,*

then $f_m(x) \leq 1 + \epsilon_s/4 \leq 1/(1 - \epsilon_s/4)$ *for each* $m \in \{1, \ldots, M\}$ *and* $g_m(x) \geq 1/(1 + \epsilon_s/4) \geq (1 - \epsilon_s/4)$ *for each* $m \in A$.

The third stopping rule is used to control the number of iterations during one phase. Here we use a parameter ω_s that depends on the phase s:

$$\omega_s = \begin{cases} \frac{2}{3M(1 - \epsilon_1/4)} & s = 1 \\[2ex] \frac{1 - \epsilon_{s-1}}{1 - \epsilon_s/4} & s > 1 \end{cases}$$

Then the third rule is defined by

$$\lambda_A(x) \leq \omega_s \lambda_{\mathcal{M}}(x^{(s-1)}) \tag{6}$$

where $x^{(s-1)}$ is the solution of phase $s - 1$ that satisfies $\lambda(x^{(s-1)}) \leq 1/(1 - \epsilon_{s-1})$.

LEMMA 8 *Let* $x^{(s-1)}$ *be the initial solution and* x *be a vector in phase* $s \geq 1$ *with* $\lambda_A(x) \leq \omega_s \lambda(x^{(s-1)})$ *for* $A \subset \mathcal{M}$. *If* $\lambda_A(x^{(s-1)}) \leq 3M/2$ *for* $s = 1$ *and* $\lambda_A(x^{(s-1)}) \leq 1/(1 - \epsilon_{s-1})$ *for* $s \geq 2$, *then we obtain* $\lambda_A(x) \leq 1/(1 - \epsilon_s/4)$.

3.3 Choice of the step length

In this subsection we describe the choice of the step length τ. We suppose that we have computed a vector x and an approximate block solution $\hat{x}$ in a phase s such that $\nu(x, \hat{x}) > t$, $p^T f(\hat{x}) \leq (1 + t) \sum_{m=1}^{M} p_m$ and $q^T g(\hat{x}) \geq \frac{1}{1+t} \sum_{m=1}^{M} q_m$ (where $t = t_s$, $p = p(x, A(x))$ and $q = q(x, A(x))$). Let $x' = (1 - \tau)x + \tau \hat{x}$. First we focus on the case where $g_m(x') < T = T(s)$ for each $m \in A(x)$. In this case we do not eliminate a component (i.e. $A(x') = A(x)$). The other case will be discussed later (in some cases we have in addition to reduce the step length). For simplification we use $\theta = \theta_{A(x)}(x)$. Since each function f_m is convex and each function g_m is concave, we get independently on the choice of τ the following inequalities

$$\theta - f_m(x') \geq (\theta - f_m(x))(1 + \tfrac{2\tau M}{t\theta} p_m(f_m(x) - f_m(\hat{x}))),$$

$$g_m(x') - 1/\theta \geq (g_m(x) - 1/\theta)(1 + \tfrac{2\tau M\theta}{t} q_m(g_m(\hat{x}) - g_m(x)))$$

for each index $m \in \mathcal{M}$ or $m \in A(x)$, respectively. We call a step length τ **feasible** if $\tau \in (0, 1)$ and if the following value:

$$\max(\max_{m \in \mathcal{M}} |\frac{2\tau M}{t\theta} p_m(f_m(x) - f_m(\hat{x}))|, \max_{m \in A(x)} |\frac{2\tau M\theta}{t} q_m(g_m(\hat{x}) - g_m(x))|)$$

$$\tag{7}$$

is bounded by $1/2$. Suppose from now on that τ is a feasible step length. Later we will specify different step lengths τ with $\tau \in (0,1)$ to obtain the bound (7). Then using $\theta - f_m(x) > 0$ and $g_m(x) - 1/\theta > 0$ we obtain $\theta - f_m(x') > 0$ and $g_m(x') - 1/\theta > 0$ for the next computed vector $x' \in B$. This implies that the objective value $\lambda_{A(x')}(x')$ for the next vector x' is at most $\theta_{A(x)}(x)$, where here $A(x') = A(x)$.

LEMMA 9 *For any two consecutive iterations in a phase with computed vectors x, x' and $A(x') = A(x)$ and any feasible step length τ, the difference $\phi_t(x, A(x)) - \phi_t(x', A(x'))$ is at least*

$$+2\tau[(p^T f(x) - p^T f(\hat{x}))/\theta + (q^T g(\hat{x}) - q^T g(x))\theta]$$
$$-\frac{4M\tau^2}{t}[(p^T f(x) + p^T f(\hat{x}))/\theta + (q^T g(\hat{x}) + q^T g(x))\theta]^2,$$

where $\theta = \theta_{A(x)}(x)$, $p = p(x, A(x))$ and $q = q(x, A(x))$.

The proof of Lemma 9 can be found in the full version. In our algorithm we use the following step lengths:

$$\tau_1 = t\nu/(4M[(p^T f(x) + p^T f(\hat{x}))/\theta + (q^T g(\hat{x}) + q^T g(x))\theta]),$$

$$\tau_2 = t\nu/(4M[(p^T f(x) + p^T f(\hat{x})) + (q^T g(\hat{x}) + q^T g(x))\theta]),$$

$$\tau_3 = t^2/(4M[(p^T f(x) + p^T f(\hat{x}))/\theta + (q^T g(\hat{x}) + q^T g(x))\theta]),$$

$$\tau_4 = t/(12M)$$

where $\theta = \theta_A(x)$ and $\nu = \nu(x, \hat{x})$. With exception of the last case, all step lengths above are feasible for any $t \in (0, 1/2]$. The last step length τ_4 is feasible only for the case $q^T g(\hat{x}) \le q^T g(x)$ and any $t \in (0, 1]$. Furthermore, note that each step length $\tau \in (0, \tau_i]$ is also feasible for $i \in \{1, \ldots, 4\}$. In our algorithm we use the step lengths (see Table 1) in dependence on the current vector x, the approximate block solution $\hat{x}$, the minimizer $\theta = \theta_{A(x)}(x)$ and the price vectors $p = p(x, A(x))$, $q = q(x, A(x))$.

The main goal now is to prove the following result. The proof can be found in the full version of the paper.

THEOREM 10 *For any two consecutive iterations in a phase with computed vectors x, x', index sets $A(x) = A(x')$ and $t \le 1/224$ we obtain: $\phi_t(x, A(x)) - \phi_t(x', A(x')) \ge \Theta(\frac{t^3}{M})$.*

3.4 Reducing the step length

Let $x' = (1 - \tau)x + \tau\hat{x}$ where x is the current vector, $\hat{x}$ is the block solution and τ is the step length as used in the previous subsection. Consider a phase

	$\theta \leq 2$	$2 < \theta \leq 200$	$200 < \theta$
$p^T f(x) - p^T f(\hat{x}) \geq 0$ and $q^T g(\hat{x}) - q^T g(x) \geq 0$	$\tau_1/120$	$\tau_1/120$	$\tau_1/2$
$p^T f(x) - p^T f(\hat{x}) < 0$ and $q^T g(\hat{x}) - q^T g(x) \geq 0$	τ_2	τ_2	τ_2
$p^T f(x) - p^T f(\hat{x}) \geq 0$ and $q^T g(\hat{x}) - q^T g(x) < 0$	$\tau_3/3$	$\tau_4/9$	$\tau_4/9$

Table 1.   The choice of the step lengths

s with threshold value $T(s)$. For simplicity we use $T = T(s)$. If $g_m(x') \leq T$ for each $m \in A(x)$, then we use x' as the next iterate and set $A(x') = \{m \in A(x)|g_m(x') < T\}$. In this case some components may be eliminated, but we use the original step length. Now we consider the case that $g_m(x') > T$ for at least one coordinate $m \in A(x)$. Let $\gamma(\tilde{\tau}) = \max_{m \in A(x)} g_m(x)(1 - \tilde{\tau}) + g_m(\hat{x})\tilde{\tau}$ for $0 \leq \tilde{\tau} \leq 1$. If $\gamma(\tau) > T$ then we reduce the step length τ. In this case we compute $\bar{\tau} < \tau$ such that $\gamma(\bar{\tau}) = T$. Using $g_m(x) < T$ for each $m \in A(x)$ and $\gamma(\tau) > T$, there is at least one component $m \in A(x)$ such that $g_m(\hat{x}) > T$. In addition, the value $\bar{\tau}$ is unique and can be computed in $O(M)$ time. We use here $x' = x(1 - \bar{\tau}) + \hat{x}\bar{\tau}$ as next iterate and set $A(x') = \{m \in A(x)|g_m(x') < T\}$. If $\gamma(\tau) \leq T$ then we do not have to reduce the step length τ and use again $x' = x(1 - \tau) + \hat{x}\tau$. But we eliminate as above all components $m \in A(x)$ with $g_m(x') \geq T$. Notice that the case with $g_m(x') > T > g_m(x)(1 - \tau) + g_m(\hat{x})\tau$ is possible (since the functions g_m are concave). For each $m \in A(x')$ we have $g_m(x') < T$. If we use a reduced step length $\bar{\tau} < \tau$ then $A(x) \neq A(x')$. But $A(x) \neq A(x')$ can happen also when $\gamma(\tau) < T$ or $g_m(x') \leq T$ for each $m \in A(x)$. Now we consider two cases depending whether we use the original step length τ or the reduced step length $\bar{\tau}$. We can prove similar to Theorem 10 the following two results (the proofs are given in the full paper).

THEOREM 11 *For any two consecutive iterations with computed vectors x, x', index sets $A(x) \neq A(x')$, $\max_{m \in A(x)} g_m(x)(1 - \tau) + g_m(\hat{x})\tau \leq T$ and $t \leq 1/224$, we obtain $\phi_t(x, A(x)) - \phi_t(x', A(x')) \geq \Theta(\frac{t^3}{M})$.*

THEOREM 12 *For any two consecutive iterations with computed vectors x, x', index sets $A(x) \neq A(x')$, $\max_{m \in A(x)} g_m(x)(1 - \tau) + g_m(\hat{x})\tau > T$ and $t \leq 1/224$, we obtain $\phi_t(x, A(x)) - \phi_t(x', A(x')) \geq 0$.*

3.5 Convex combination of different vectors

First we can prove an upper bound for the packing constraints.

LEMMA 13 *For any iteration of the phase s with computed vector x, $\lambda_f(x) = \max_{1 \leq m \leq M} f_m(x)$ is bounded by $4M/t_s$.*

Lemma 13 shows that the values $f_m(x)$ are not arbitrary large in the algorithm. Notice that this is independently from the chosen step length $\tau \in (0, 1)$. We use this bound for the convex combination below. Notice that in addition the components $p_m(x, A(x))$ of the price vector $p(x, A(x))$ are not arbitrary small (i.e. $p_m(x, A(x)) > t_s/(2M)$). At the end of phase s we have computed a vector $y \in B$ with $\lambda_{A(y)}(y) \leq 1/(1 - \epsilon_s/4)$. This implies $f_m(y) \leq 1/(1 - \epsilon_s/4)$ for each $m \in \{1, \ldots, M\}$ and $g_m(y) \geq 1 - \epsilon_s/4$ for each $m \in A(y)$. The goal is now to compute a vector $x^{(s)} \in B$ with $\lambda_{\mathcal{M}}(x^{(s)}) \leq 1/(1 - \epsilon_s)$. The key idea is to use a convex combination over several vectors computed during the phase. Let $x_1, \ldots, x_k$ be the vectors in phase s where at least one function g_m is eliminated (i.e. where $g_m(x_i) \geq T(s)$). Clearly, $k \leq M$. We have $x_1 = x^{(s-1)}$ if $g_m(x^{(s-1)}) \geq T(s)$ for at least one $m \in \mathcal{M}$ (here $x^{(s-1)}$ is the solution of the previous phase). We take the following convex combination:

$$x^{(s)} = \sum_{i=1}^{k} \frac{\epsilon_s^2}{264 M^2} x_i + \left(1 - \frac{k \epsilon_s^2}{264 M^2}\right) y.$$

Since the set B is convex and $x_1, \ldots, x_k, y \in B$, we obtain $x^{(s)} \in B$. Our threshold value $T(s)$ is equal to $528\left(\frac{M^3}{\epsilon_s^2}\right) \cdot \frac{1}{\lambda_{\mathcal{M}}(x^{(s-1)})}$. Notice that $T(s) \leq 528 M^3/\epsilon_s^2$, since $\lambda_{\mathcal{M}}(x^{(s-1)}) \geq 1$ (otherwise we are done). Then we can prove:

LEMMA 14 *The computed solution $x^{(s)}$ satisfies $\lambda_{\mathcal{M}}(x^{(s)}) \leq 1/(1 - \epsilon_s)$.*

4 Analysis of the approximation algorithm

In this subsection we determine the total number of iterations of our algorithm. To do this we calculate first the number of iterations N_s in a single phase s. Let $y, \tilde{y}$ denote the initial and final iterate of phase s. Furthermore, let $\bar{y}$ be the solution after $\bar{N}_s = N_s - 1$ iterations. For consecutive iterations with computed vectors x, x' in a phase and $A(x) = A(x')$, the difference in the potential values $\phi_t(x, A(x)) - \phi_t(x', A(x')) \geq \frac{ct^3}{M}$ where c is a positive constant and $t = t_s = \epsilon_s/32$. In addition, there are at most M iterations with consecutive vectors x, x' and different subsets $A(x) \neq A(x')$ (i.e. in these iterations at least one component is eliminated). In these cases, we have $\phi_t(x, A(x)) - \phi_t(x', A(x')) \geq 0$. Therefore, $\phi_t(y, A(y)) - \phi_t(\bar{y}, A(\bar{y})) \geq \frac{ct^3}{M}(\bar{N}_s - M)$. Then we can prove the following result:

THEOREM 15 *The number of iterations N_s in phase s is at most*

$$O(M\epsilon_s^{-2}\ln(M\epsilon_s^{-1}))$$

and the total number of iterations of our algorithm is at most

$$O(M\epsilon^{-2}\ln(M\epsilon^{-1})).$$

Remark: The root $\theta_A(x)$ can often be computed only approximately, but an accuracy of $O(\epsilon^2/M)$ for $\theta_A(x)$ is sufficient to generate the above bounds on the number of iterations. With this required accuracy, the number of evaluations of the sum $\frac{t\theta}{M}\sum_{m=1}^{M}\frac{1}{\theta-f_m(x)} + \frac{t}{M\theta}\sum_{m\in A}\frac{1}{g_m(x)-1/\theta}$ is bounded by $O(\ln(M\epsilon^{-1}))$. This gives $O(M\ln(M\epsilon^{-1}))$ arithmetic operations to determine $\theta_A(x)$ approximately.

5   Concluding Remarks

In this paper we have presented an approximation algorithm for the mixed packing and covering problem that uses only $O(M\epsilon^{-2}\ln(M\epsilon^{-1}))$ calls to an oracle of the form: compute a $\hat{x}\in B$ such that $f(x)\leq 1$ and $g(x)\geq 1$. We note that probably the computation of the convex combination can be avoided and the number of calls to the oracle can be improved to $O(M(\ln M + \epsilon^{-2}\ln\epsilon^{-1}))$. The details will be given in the full paper.

References

[1] Bienstock, D. (2002). Potential function methods for approximately solving linear programming problems: Theory and practive. Boston: Kluwer.

[2] Charikar,M., Chekuri, C., Goel, A., Guha, S., and Plotkin, S. A. (1998). Approximating a finite metric by a small number of tree metrics, *Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science*, 379-388.

[3] Caragiannis, I., Ferreira, A., Kaklamanis, C., Perennes, S., and Rivano, H. (2001). Fractional path coloring with applications to WDM networks, *Proceedings 28th International Colloquium on Automata, Languages, and Programming*, ICALP 2001, LNCS 2076, 732-743.

[4] Fleischer, L. (2004). A fast approximation scheme for fractional covering problems with variable upper bounds, *Proceedings of the 15th ACM-SIAM Symposium on Discrete Algorithms*, SODA 2004.

[5] Garg, N. and Könemann, J. (1998). Fast and simpler algorithms for multicommodity flow and other fractional packing problems, *Proceedings of the 39th IEEE Annual Symposium on Foundations of Computer Science*, FOCS 1998, 300-309.

[6] Grigoriadis, M.D. and Khachiyan, L.G. (1994). Fast approximation schemes for convex programs with many blocks and coupling constraints, *SIAM Journal on Optimization*, 4: 86-107.

[7] Grigoriadis, M.D. and Khachiyan, L.G. (1996). Coordination complexity of parallel price-directive decomposition, *Mathematics of Operations Research*, 2: 321-340.

[8] Grigoriadis, M.D., Khachiyan, L.G., Porkolab, L., and Villavicencio, J. (2001). Approximate max-min resource sharing for structured concave optimization, *SIAM Journal on Optimization*, 41: 1081-1091.

[9] Jansen, K. and Porkolab, L. (2002). On preemptive resource constrained scheduling: polynomial-time approximation schemes, *Proceedings of the 9th International Conference on Integer Programming and Combinatorial Optimization*, IPCO 2002, LNCS 2337, 329-349.

[10] Jansen, K. and Zhang, H. (2002) Approximation algorithms for general packing problems with modified logarithmic potential function, *Proceedings of the 2nd IFIP International Conference on Theoretical Computer Science*, TCS 2002, Foundations of information technology in the era of network and mobile computing, Kluwer Academic Publisher, 2002, 255-266.

[11] Jansen, K. (2004). Approximation algorithms for the general max-min resource sharing problem: faster and simpler, to appear in: *Proceedings of the 9th Scandinavian Workshop on Algorithm Theory*, SWAT 2004.

[12] Kenyon, C. and E. Remila, E. (1996). Approximate strip packing, *Proceedings 37th IEEE Symposium on Foundations of Computer Science*, FOCS 1996, 31-36.

[13] Könemann, J. (2000). Fast combinatorial algorithms for packing and covering problems, Diploma Thesis, Max-Planck-Institute for Computer Science Saarbrücken.

[14] Plotkin, S.A., Shmoys, D.B., and Tardos, E. (1995). Fast approximation algorithms for fractional packing and covering problems, *Mathematics of Operations Research*, 20: 257-301.

[15] Villavicencio, J. and Grigoriadis, M.D. (1997). Approximate Lagrangian decomposition with a modified Karmarkar logarithmic potential, *Network Optimization*, Lecture Notes in Economics and Mathematical Systems, 450: 471-485.

[16] Young, N.E. (1995). Randomized rounding without solving the linear program, *Proceedings of the 6th ACM-SIAM Symposium on Discrete Algorithms* SODA 1995, 170-178.

[17] Young, N.E. (2001). Sequential and parallel algorithms for mixed packing and covering, *Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science*, FOCS 2001, 538-546.